COMP9444 Neural Networks and Deep Learning

Quiz 5 (Recurrent Networks)

This is an optional quiz to test your understanding of the material from Week 5.

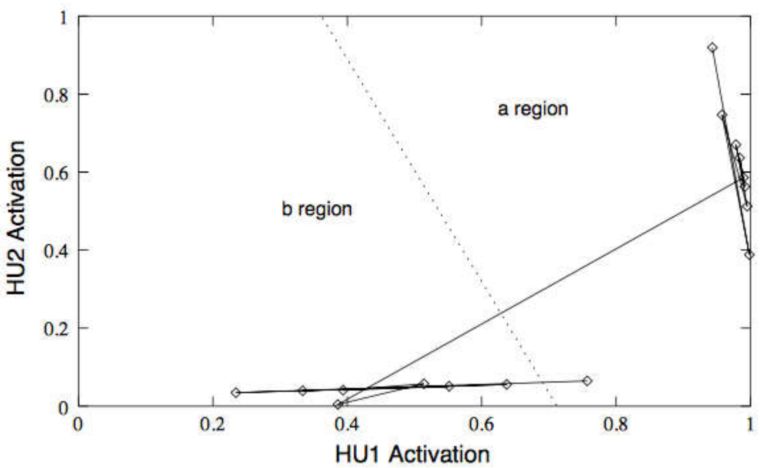
1. Explain the format and method by which input was fed to the NetTalk system, and the target output.

    Characters were fed to NetTalk using a sliding window approach. The characters in a 7-word window were encoded with a 1-hot encoding to form the input of size 7 × 29. The network had 26 outputs - each corresponding to a letter of the phonetic alphabet. The target output was the correct pronunciation of the central character in the input.

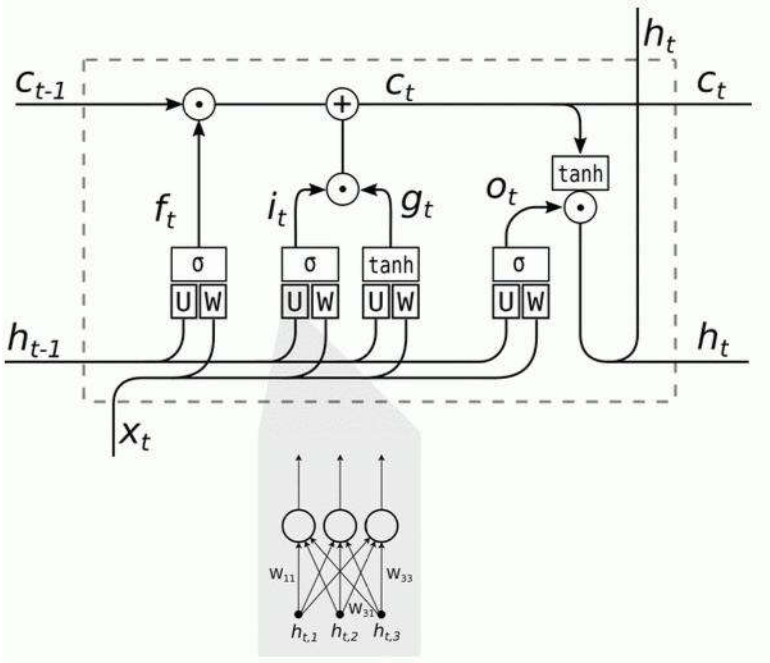
2. Explain the role of the context layer in an Elman network.

    The context layer is a copy of the hidden layer at the previous timestep. The hidden layer accepts connections from both the hidden and context layers. This in theory allows the network to retain "state" information for an indefinite period of time.

3. Draw a diagram showing the hidden unit activations of a Simple Recurrent Network with two hidden units trained on the $a^n b^n$ task, as it processes $a^8 b^8$.



4. Draw a diagram of an LSTM and write the equations for its operation.

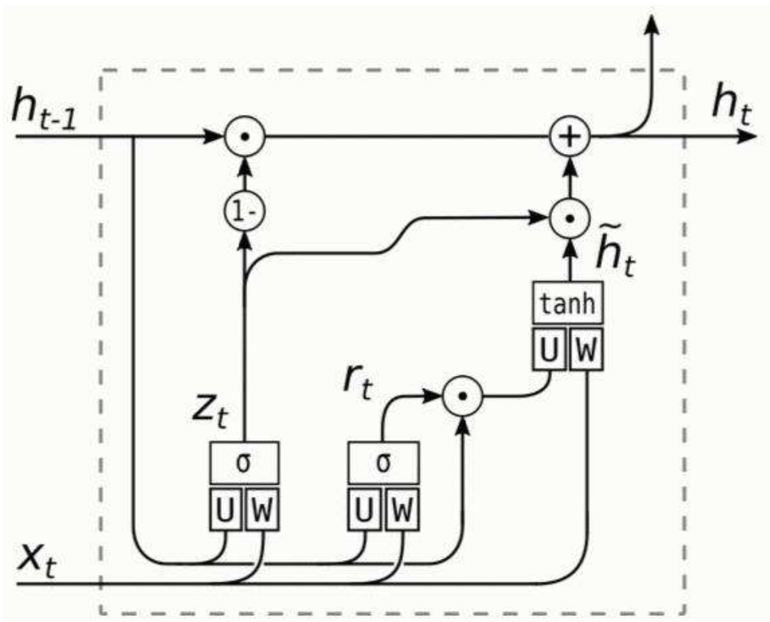


Gates:
$f_t = \sigma(W_f\, x_t + U_f\, h_{t-1} + b_f)$ [forget gate]
$i_t = \sigma(W_i\, x_t + U_i\, h_{t-1} + b_i)$ [input gate]
$g_t = \tanh(W_g\, x_t + U_g\, h_{t-1} + b_g)$
$o_t = \sigma(W_o\, x_t + U_o\, h_{t-1} + b_o)$ [output gate]

State:
$c_t = c_{t-1} \otimes f_t + i_t \otimes g_t$

Output:
$h_t = \tanh(c_t) \otimes o_t$

5. Draw a diagram of a Gated Recurrent Unit and write the equitions for its operation.



Gates:
$z_t = \sigma(W_z\, x_t + U_z\, h_{t-1} + b_z)$
$r_t = \sigma(W_r\, x_t + U_r\, h_{t-1} + b_r)$

Candidate Activation:
$\hat{h}_t = \tanh(W x_t + U(r_t \otimes h_{t-1}) + b_h)$

Output:
$h_t = (1 - z_t) \otimes h_{t-1} + z_t \otimes \hat{h}$

6. Briefly describe the problem of *long range dependencies*, and discuss how well each of the following architectures is able to deal with long range dependencies:
   a. sliding window approach
   b. Simple Recurrent (Elman) Network
   c. Long Short Term Memory (LSTM)
   d. Gated Recurrent Unit (GRU)

   For sequence processing tasks, it can happen that the correct output depends on inputs that occurred many timesteps earlier. The sliding window approach is unable to take account of any input beyond the edge of the window. Simple Recurrent Networks can learn medium-range dependencies but may struggle with long range dependencies unless the training data are carefully constructed and the amount of "state" information is limited. LSTMs and GRUs are more successful at learning long range dependencies because they can learn to use some dimensions for short-term processing and others for long-term information.
